Comparison of Standard and Pretrained CNN Models for Potato, Cotton, Bean and Banana Disease Detection

Soner KIZILOLUK[1*]

[1] Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Malatya Turgut Özal University, Malatya, Turkey.

ABSTRACT: Plant diseases lead to a significant decrease in product efficiency and economic losses for producers. However, early detection of plant diseases plays an important role in preventing these losses. Today, Convolutional Neural Network (CNN) models are widely used for image processing in many fields such as face recognition, climate, health, and agriculture. But in these models, the weights of the layers are randomly initialized during training, which increases training time and decreases performance. With the method known as Transfer Learning in the literature, CNN models are trained on large databases such as ImageNet. Then, pretrained CNN models are created using the weights obtained in this training. Thus, training time decreases while performance improves. In this study, standard and pretrained versions of popular CNN models DarkNet-19, GoogleNet, Inception-v3, Resnet-18, and ShuffleNet have been used for automatic classification of diseases from leaf images of potato, cotton, bean, and banana. In the experimental study, the classification performances of all these standard and pretrained CNN models are presented comparatively. Experimental results have shown that the performance of CNN models is significantly improved by transfer learning, even in a small number of epochs.

Keywords: Convolutional Neural Network, Transfer Learning, Disease Detection, Classification.

1. INTRODUCTION

In parallel with the rapid increase in the world's population, the demand for agricultural products also increases. Increased productivity in agricultural products is very important to meet this demand. However, plant diseases lead to decreases in crop productivity, economic losses, and an inability to meet the increasing demand. However, early detection of plant diseases plays an important role in preventing these losses. Experts and laboratories are required to detect plant diseases. However, this increases the transaction costs and leads to time loss [1]. Today, methods based on image processing technology are frequently used for the detection of plant diseases [2]. Convolutional Neural Network (CNN) models, one of the most up-to-date image processing technologies, are widely used in many areas such as face recognition, climate, health, and agriculture.

There are many CNN-based studies for the detection of plant diseases in the literature. Chen et al. [3] proposed a CNN model called Mobile-Atten, which is based on the MobileNet-v2 CNN model. In the study, they performed disease detection and classification using rice leaves with the Mobile-Atten model. In another study focusing on disease detection using rice leaves,

*Corresponding Author: soner.kiziloluk@ozal.edu.tr
ORCID number of authors: [1] 0000-0002-0381-9631

Shrivastava et al. [4] used AlexNet architecture. In the study, they reached a 91.37% classification accuracy rate. Abbas et al. [5] proposed a method called C-GAN based on the DenseNet CNN model for disease detection using tomato leaves and achieved a 99.51% success rate. Türkoğlu et al. [1] suggested a new CNN model consisting of 18 layers for disease detection using apricot images. Sert [6] proposed a new method called Faster R-CNN-GC, which is based on image compositing, GoogleNet, and Faster R-CNN. In the study, the detection of pepper and potato leaves was carried out and the type of disease was determined from these leaves. On the other hand, Khan et al. [7] proposed an approach called CCDF for disease detection using apple and banana leaf images. In this approach, feature extraction from apple and banana images is performed with pretrained VGG-16 and AlexNet models, and then classification is performed with the Support Vector Machine (SVM). Aksoy et al. [8] performed disease detection with AlexNet, DenseNet-121, Resnet-34, Squeezenet, and VGG-16 CNN models using apple leaf images and compared the performances of these models. Hassan et al.[9]performed disease detection with Inception-v3, InceptionResNet-v2, MobileNet-v2, and EfficientNet-b0 CNN models using leaf images of 14 different plants. Arivazhagan and League [10] proposed a 7-layer CNN model for disease detection using leaf images of the mango plant. The proposed model was found to classify the test data correctly at a rate of 96.67%. Priyadharshini et al. [11] performed disease detection using corn leaf images with the modified LeNet model and achieved a success rate of 97.89%. Dinata et al. [12] achieved a 63.7% success rate with the CNN model they proposed for disease detection using strawberry leaf images.

In this study, DarkNet-19, GoogleNet, Inception-v3, Resnet-18, and ShuffleNet models, which are popular CNN models, have been used for automatic classification of diseases from leaf images of potato, cotton, bean, and banana. In addition, pretrained versions of these models have also been used in the study. Experiment results show that the performance is significantly increased even in a small number of epochs with "Transfer Learning."

The rest of the study has been organized as follows. In the Material and Methods section, CNN models, the concept of transfer learning, the datasets, and the workflow of the study are presented. In the Results and Discussion section, experimental study and results are given in detail. In the last section, the obtained results are discussed.

2. MATERIAL AND METHODS

2.1. Convolutional Neural Networks (CNN)

CNN is a special type of Neural Networks (NN). It was developed by drawing inspiration from the biological model of the visual cortex of animals. CNNs are frequently used in image and audio processing due to advantages such as highly accurate identification, automatic feature extraction, and learning [13]. CNNs consist of many layers. Its basic layers are the convolution layer, pooling layer, and classification layer. In the convolution layer, feature extraction is carried out by hovering small filters over the image. The pooling layer is used to reduce the size of the features. The final layer of CNN and the layer in which classification is carried out is the classification layer [1].

In this study, DarkNet-19, GoogleNet, Inception-v3, Resnet-18 and, ShuffleNet CNN models have been used. The DarkNet-19 model was proposed by Redmon [14]. It consists of 19 layers in total and the image input size is 256. The GoogleNet model was proposed by Szegedy et al. [15] and the image input size and number of layers are 224 and 22, respectively. In the Inception-v3 model proposed by Szegedy et al. [16], the image input size is 299 and it consists of 48 layers. The ResNet-18 model with an image input size of 224 and 18 layers was proposed

by He et al. [17]. In the ShuffleNet model proposed by Zhang et al. [18], the image input size is 224 and it consists of 50 layers.

In this study, pretrained versions of CNN models created with the Transfer Learning method have also been used. In CNN models, the weights in the layers are randomly initialized during training. This leads to a longer training period and decreased performance. Transfer Learning is a learning approach that examines the use of the information learned during the training phase of machine learning systems in different or similar problem solutions [19]. With the Transfer Learning method, CNN models are first trained on large databases such as ImageNet. Then, pretrained CNN models are created using the weights obtained in this training. Thus, training time decreases while performance improves.

2.2. Image Datasets

In this study, 4 different datasets on potato, cotton, bean, and banana, obtained from Kaggle, have been used [20-23]. The potato dataset consists of 3 classes, Early Blight, Healthy, and Late Blight, and 2152 images. The cotton dataset consists of a total of 4 classes, bacterial blight, curl virus, fusarium wilt, and healthy, as well as 1711 images. The bean dataset consists of 3 classes, angular leap spot, bean rust, and healthy, as well as 1296 images. The banana dataset consists of 4 classes, Cordana, Healthy, Pestalotiopsis, and Sigatoka, as well as 936 images. Samples images from both datasets are shown in Figure 1.
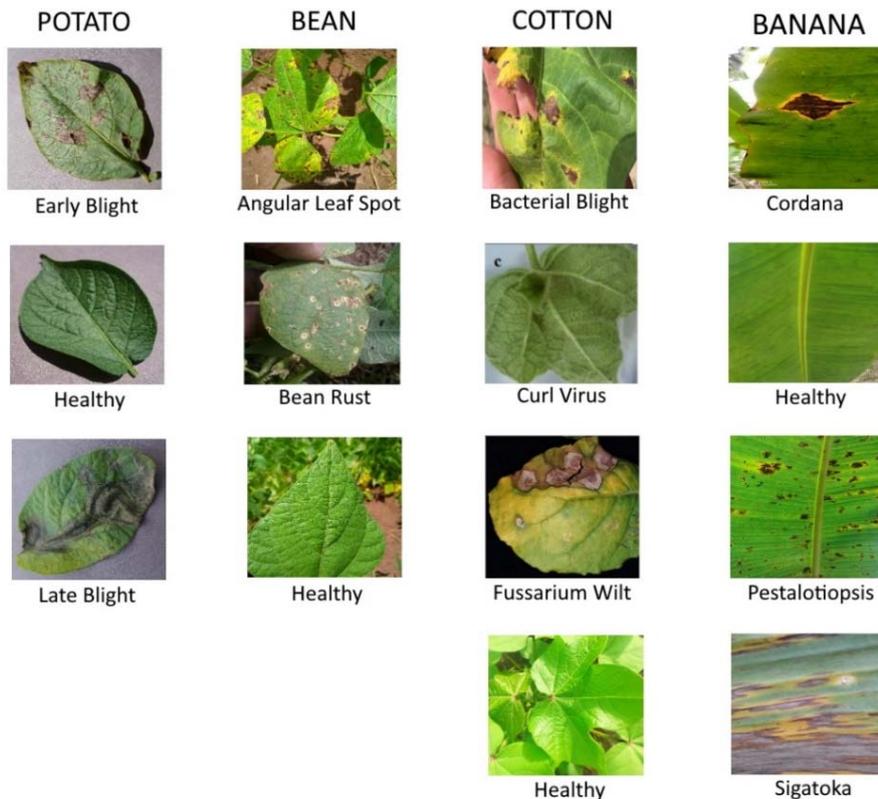
Figure 1. Sample images from datasets

2.3. Methods

In this study, standard and pretrained versions of DarkNet-19, GoogleNet, Inception-v3, Resnet-18, and ShuffleNet CNN models have been used for automatic classification of diseases using leaf images of potato, cotton, bean, and banana. In all datasets, %80 and %20 images have been used for training and testing processes, respectively. In addition, 10% of the training data have been selected for train validation. The workflow diagrams of the study have been given in Figure 2 for standard CNN models and in Figure 3 for pre-trained CNN models. In standard CNN models, the weights of the layers have been initialized randomly during training. On the other hand, In pretrained CNN models, standard CNN models are first trained on large databases such as ImageNet. Then the weights of the layers obtained from the training are transferred.
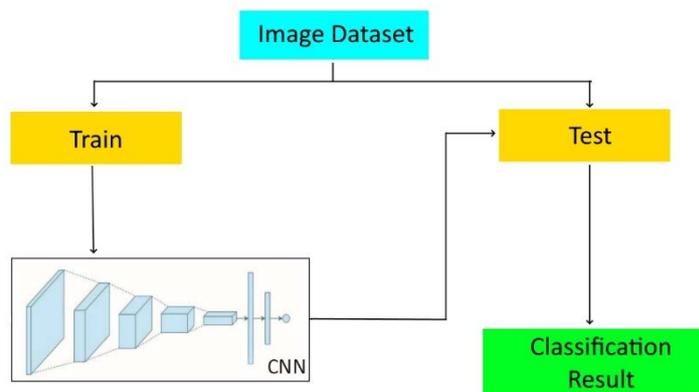
Figure 2. Workflow for standard CNN models
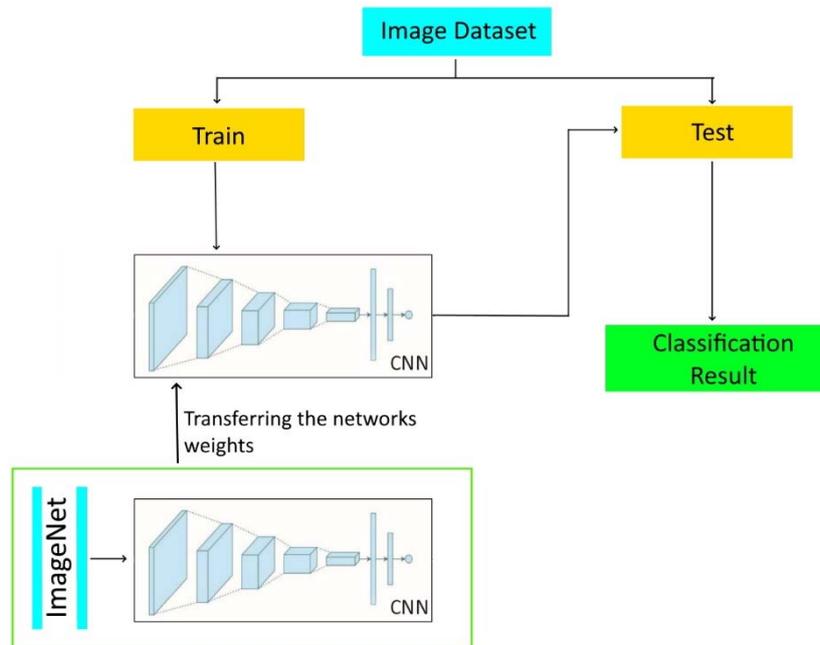
Figure 3. Workflow for pretrained CNN models

3. RESULTS AND DISCUSSION

For all CNN models, the max epoch value, the mini-batch size value, and the initial learning rate value have been selected as 5, 25, and 0.001, respectively. In addition, all CNN models have been run ten times on all datasets, and comparison has been carried according to the mean and maximum accuracy values obtained in train validation and testing processes. Additionally, mean training accuracy curves, confusion matrices, and ROC curves of each CNN model have been presented comparatively.

Table 1 shows the results obtained from the potato dataset. When the results are examined, it is seen that Pretrained-DarkNet-19 had the best performance with 99.59 mean accuracy and 100 maximum accuracy values for training validation. For the test, Pretrained-ResNet-18 had the best results with 99.67 mean accuracy and 100 maximum accuracy. Standard GoogleNet had the worst performance for both training validation and testing. In addition, it is easily seen that the performance of standard CNN models has increased by up to 7% with the transfer learning method.

Table 1. Results for the Potato dataset

	TRAIN VALIDATION			TEST		
	Acc. (mean)	Acc. (max)	Std.Dev.	Acc. (mean)	Acc. (max)	Std.Dev.
DarkNet-19	98.63	99.13	0.48	98.49	99.30	0.55
GoogleNet	92.65	96.51	3.66	93.14	96.05	2.44
Inception-v3	95.70	97.38	1.26	95.67	96.98	0.86
ResNet-18	98.20	99.13	0.65	97.44	97.91	0.49
ShuffleNet	94.39	96.80	1.25	94.37	96.74	1.45
Pretrained-DarkNet-19	**99.59**	**100.00**	0.46	99.56	**100.00**	0.43
Pretrained-GoogleNet	99.01	99.42	0.53	98.81	99.53	0.79
Pretrained-Inception-v3	99.10	99.42	0.29	99.30	99.77	0.36
Pretrained-ResNet-18	99.45	**100.00**	**0.29**	**99.67**	**100.00**	**0.22**
Pretrained-ShuffleNet	98.90	99.71	0.49	99.12	99.77	0.46

Table 2 shows the results obtained from the cotton dataset. When the results are examined, it is seen that Pretrained-GoogleNet had the best results in all values except mean accuracy for the test. In addition, it is easily seen that the performance of standard CNN models has increased by around 25% with the transfer learning method.

Table 3 shows the results obtained from the Bean dataset. When the results are examined, it is seen that Pretrained-DarkNet-19 had the best performance with 94.54 mean accuracy and 97.1 maximum accuracy for training validation, and 94.22 mean accuracy and 96.9 maximum accuracy for testing. In addition, it is easily seen that the performance of standard CNN models has increased by 20% with the transfer learning method.

Table 2. Results for the Cotton dataset

	TRAIN VALIDATION				TEST	
	Acc. (mean)	Acc. (max)	Std.Dev.	Acc. (mean)	Acc. (max)	Std.Dev.
DarkNet-19	71.46	77.01	3.43	73.56	80.76	4.51
GoogleNet	66.24	72.99	4.98	65.34	74.93	6.54
Inception-v3	69.23	74.45	3.01	63.85	77.55	5.82
ResNet-18	75.69	80.66	3.80	75.10	79.59	4.03
ShuffleNet	70.11	74.45	3.59	65.25	73.47	5.24
Pretrained-DarkNet-19	71.28	87.96	12.83	69.59	87.76	10.58
Pretrained-GoogleNet	**92.85**	**95.62**	2.52	91.05	**94.17**	1.91
Pretrained-Inception-v3	89.60	91.24	0.99	**91.95**	93.88	**1.49**
Pretrained-ResNet-18	89.89	90.88	**0.89**	89.42	93.29	2.06
Pretrained-ShuffleNet	88.21	90.51	1.76	89.10	91.84	1.85

Table 3. Results for the Bean dataset

	TRAIN VALIDATION				TEST	
	Acc. (mean)	Acc. (max)	Std.Dev.	Acc. (mean)	Acc. (max)	Std.Dev.
DarkNet-19	75.27	77.78	2.71	74.61	79.46	2.33
GoogleNet	69.66	74.40	3.38	65.08	69.38	2.69
Inception-v3	71.11	74.40	2.08	68.80	71.71	2.89
ResNet-18	72.90	78.26	2.93	71.32	76.36	3.78
ShuffleNet	65.02	69.57	4.37	63.37	67.44	3.93
Pretrained-DarkNet-19	**94.54**	**97.10**	1.55	**94.22**	**96.90**	1.84
Pretrained-GoogleNet	91.01	92.27	2.01	91.36	93.41	1.89
Pretrained-Inception-v3	91.93	92.75	**0.60**	93.37	94.19	**0.62**
Pretrained-ResNet-18	94.30	96.14	1.01	94.69	96.51	1.14
Pretrained-ShuffleNet	92.13	94.20	1.64	91.82	94.19	1.28

Table 4. Results for the Bean dataset

	TRAIN VALIDATION				TEST	
	Acc. (mean)	Acc. (max)	Std.Dev.	Acc. (mean)	Acc. (max)	Std. Dev.
DarkNet-19	89.47	92.67	3.45	87.82	93.09	3.68
GoogleNet	69.87	81.33	10.19	70.90	81.38	7.97
Inception-v3	82.80	86.00	3.40	81.65	86.70	4.47
ResNet-18	87.40	90.67	3.96	85.64	91.49	3.82
ShuffleNet	76.87	82.67	4.66	75.96	85.11	5.88
Pretrained-DarkNet-19	92.73	98.00	4.47	89.31	94.15	2.69
Pretrained-GoogleNet	**96.60**	**98.67**	**1.06**	**93.30**	**95.21**	1.65
Pretrained-Inception-v3	94.33	96.00	1.23	91.38	93.09	**1.25**
Pretrained-ResNet-18	94.33	97.33	1.78	90.59	93.09	2.08
Pretrained-ShuffleNet	92.53	94.67	1.36	89.26	92.55	2.09

Table 4 shows the results obtained from the Banana dataset. When the results are examined, it is seen that Pretrained-GoogleNet had the best performance. In addition, it is easily seen that the performance of standard CNN models has increased by around 10% with the transfer learning method.

Mean training accuracy curves of all models obtained from the Potato, Cotton, Bean, and Banana datasets are shown in Figure 4, Figure 5, Figure 6, and Figure 7, respectively. When the figures are examined, it is seen that the accuracy values of the pretrained CNN models reach high values even in the early iteration. In standard CNN models, it is seen that accuracy values do not increase much even in later iterations.
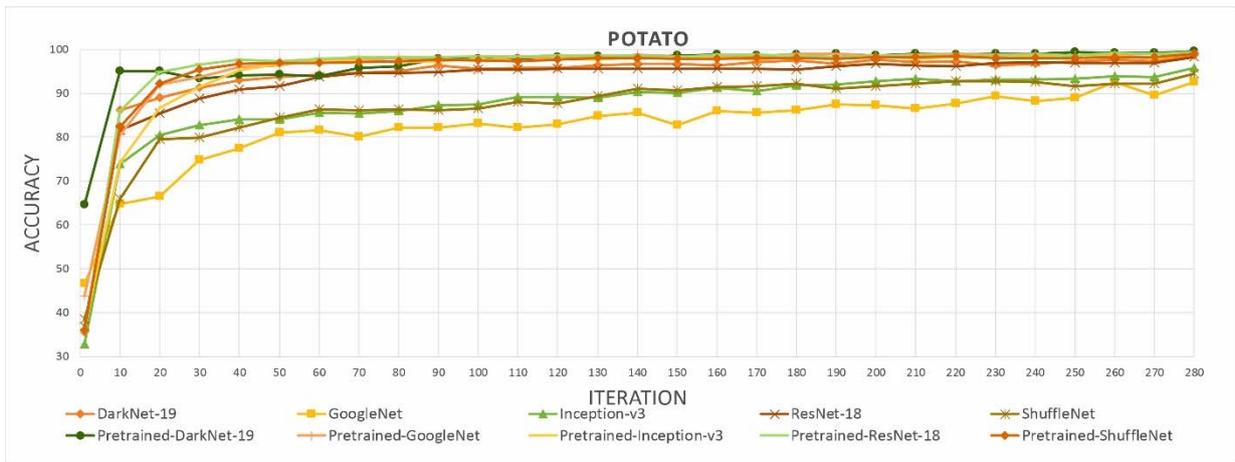
Figure 4. Mean training accuracy curves for the Potato dataset
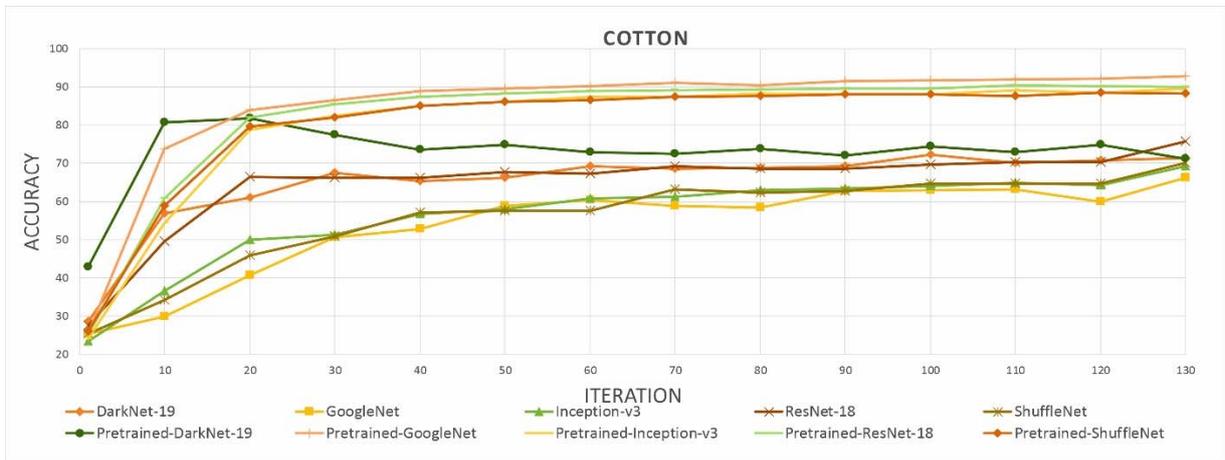
Figure 5. Mean training accuracy curves for the Cotton dataset
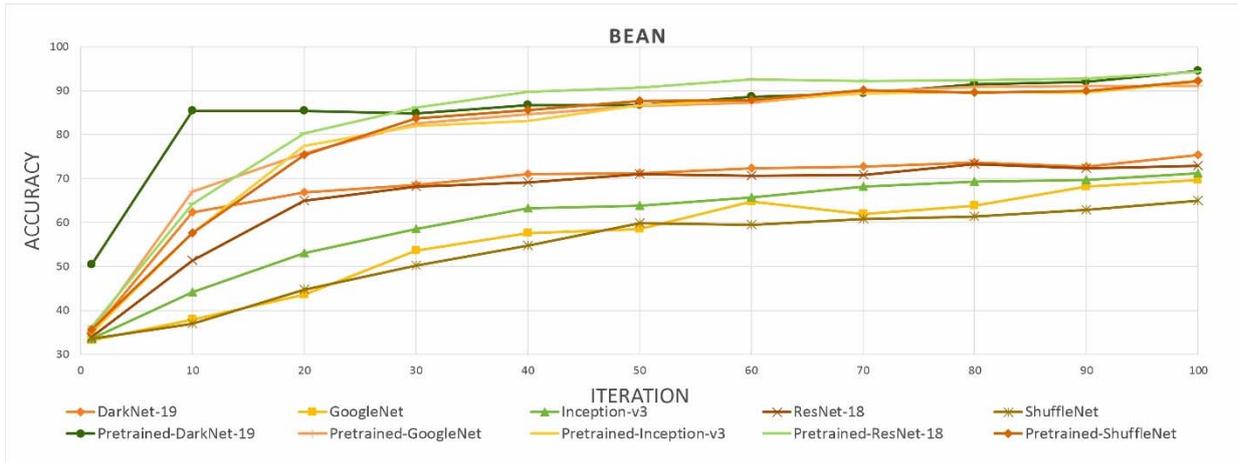
Figure 6. Mean training accuracy curves for the Bean dataset
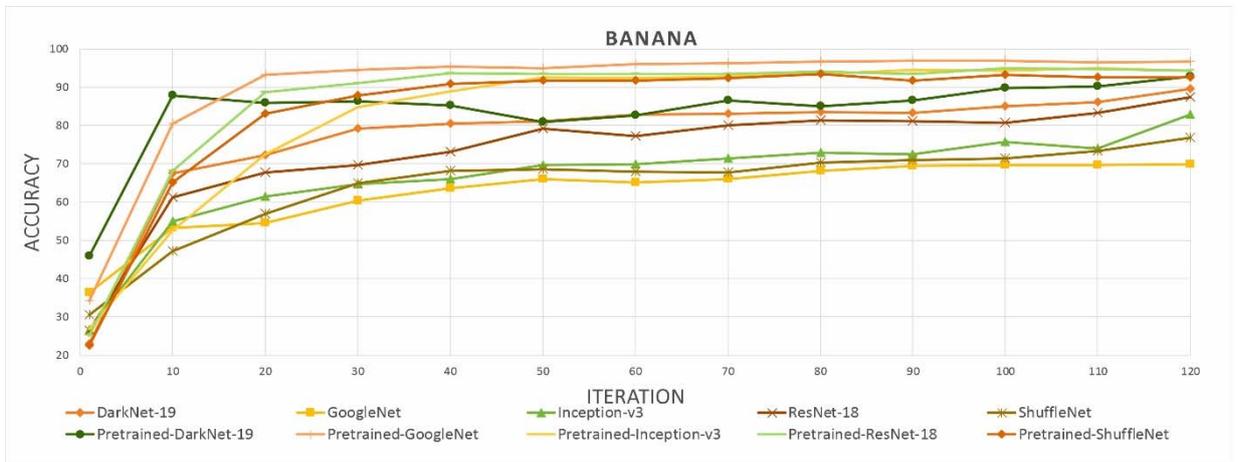
Figure 7. Mean training accuracy curves for the Banana dataset

Confusion matrices obtained from the tests of potato, cotton, bean and, banana datasets in all CNN models are given in Figures 8, 9, 10, and 11, respectively. A confusion matrix is a table that is used to describe the performance of a model by referring to its accuracy rates in each class. The rows in the confusion matrix show the predicted class (Output Class) and the columns show the true class (Target Class). When the matrixes are examined, it is easily seen that the class-based correct prediction rates of the pretrained CNN models are higher than the standard CNN models.

DarkNet-19

	E. Blight	Healthy	L. Blight	
E. Blight	198.20	0.00	0.40	99.80
Healthy	0.00	27.90	2.20	92.69
L. Blight	1.80	2.10	197.40	98.06
	99.10	93.00	98.70	98.49

GoogleNet

	E. Blight	Healthy	L. Blight	
E. Blight	191.80	0.00	2.00	98.97
Healthy	0.10	13.00	2.30	84.42
L. Blight	8.10	17.00	195.70	88.63
	95.90	43.33	97.85	93.14

Inception-v3

	E. Blight	Healthy	L. Blight	
E. Blight	195.10	0.60	6.00	96.73
Healthy	0.20	25.30	3.00	88.77
L. Blight	4.70	4.10	191.00	95.60
	97.55	84.33	95.50	95.67

ResNet-18

	E. Blight	Healthy	L. Blight	
E. Blight	197.20	0.20	1.60	99.10
Healthy	0.00	27.30	3.90	87.50
L. Blight	2.80	2.50	194.50	97.35
	98.60	91.00	97.25	97.44

ShuffleNet

	E. Blight	Healthy	L. Blight	
E. Blight	191.70	1.10	6.40	96.23
Healthy	0.00	22.90	2.40	90.51
L. Blight	8.30	6.00	191.20	93.04
	95.85	76.33	95.60	94.37

Pretrained DarkNet-19

	E. Blight	Healthy	L. Blight	
E. Blight	199.60	0.00	0.30	99.85
Healthy	0.00	30.00	1.20	96.15
L. Blight	0.40	0.00	198.50	99.80
	99.80	100.00	99.25	99.56

Pretrained GoogleNet

	E. Blight	Healthy	L. Blight	
E. Blight	199.70	0.00	1.40	99.30
Healthy	0.00	28.20	1.60	94.63
L. Blight	0.30	1.80	197.00	98.95
	99.85	94.00	98.50	98.81

Pretrained Inception-v3

	E. Blight	Healthy	L. Blight	
E. Blight	200.00	0.10	0.50	99.70
Healthy	0.00	28.20	0.70	97.58
L. Blight	0.00	1.70	198.80	99.15
	100.00	94.00	99.40	99.30

Pretrained ResNet-18

	E. Blight	Healthy	L. Blight	
E. Blight	200.00	0.00	0.20	99.90
Healthy	0.00	29.80	1.00	96.75
L. Blight	0.00	0.20	198.80	99.90
	100.00	99.33	99.40	99.67

Pretrained ShuffleNet

	E. Blight	Healthy	L. Blight	
E. Blight	198.70	0.10	0.50	99.70
Healthy	0.00	29.00	1.00	96.67
L. Blight	1.30	0.90	198.50	98.90
	99.35	96.67	99.25	99.12

Figure 8. Confusion matrices of Potato dataset

DarkNet-19

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	67.20	7.90	8.40	20.9	64.37
C. Virus	13.40	67.50	1.10	19.5	66.50
F. Wilt	6.60	3.30	74.50	1.5	86.73
Healthy	2.8	5.3	0	43.10	84.18
	74.67	80.36	88.69	50.71	73.56

GoogleNet

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	53.30	11.20	7.70	11.1	63.99
C. Virus	19.60	57.40	4.40	28.9	52.04
F. Wilt	16.70	2.30	71.90	3.5	76.17
Healthy	0.4	13.1	0	41.50	75.45
	59.22	68.33	85.60	48.82	65.34

Inception-v3

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	47.40	18.60	5.40	16.5	53.92
C. Virus	29.20	55.60	6.40	23.1	48.64
F. Wilt	7.00	4.90	72.20	1.6	84.25
Healthy	6.4	4.9	0	43.80	79.49
	52.67	66.19	85.95	51.53	63.85

ResNet-18

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	67.90	9.20	4.30	12.6	72.23
C. Virus	15.90	64.50	5.60	19.5	61.14
F. Wilt	4.50	3.50	73.90	1.6	88.50
Healthy	1.7	6.8	0.2	51.30	85.50
	75.44	76.79	87.98	60.35	75.10

ShuffleNet

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	46.90	15.70	6.90	13.2	56.71
C. Virus	29.40	57.80	4.40	24.4	49.83
F. Wilt	10.80	4.70	72.70	1	81.50
Healthy	2.9	5.8	0	46.40	84.21
	52.11	68.81	86.55	54.59	65.25

Pretrained DarkNet-19

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	69.70	7.20	8.20	22.6	64.72
C. Virus	8.80	64.50	12.70	12.7	65.35
F. Wilt	5.80	6.20	59.30	4.5	78.23
Healthy	5.7	6.1	3.8	45.20	74.34
	77.44	76.79	70.60	53.18	69.59

Pretrained GoogleNet

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	84.10	0.40	5.00	3.2	90.72
C. Virus	2.70	72.50	0.80	3.3	91.42
F. Wilt	1.30	8.80	78.20	1	87.57
Healthy	1.9	2.3	0	77.50	94.86
	93.44	86.31	93.10	91.18	91.05

Pretrained Inception-v3

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	86.40	0.00	3.70	0	95.89
C. Virus	2.60	74.20	1.60	1.6	92.75
F. Wilt	0.00	9.70	78.40	7	82.44
Healthy	1	0.1	0.3	76.40	98.20
	96.00	88.33	93.33	89.88	91.95

Pretrained ResNet-18

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	84.40	0.20	3.10	4.5	91.54
C. Virus	2.00	73.70	3.20	6.5	86.30
F. Wilt	2.50	10.10	77.50	2.9	83.33
Healthy	1.1	0	0.2	71.10	98.20
	93.78	87.74	92.26	83.65	89.42

Pretrained ShuffleNet

	B. Blight	C. Virus	F. Wilt	Healthy	
B. Blight	76.80	1.30	2.10	0.8	94.81
C. Virus	6.20	72.50	2.80	3	85.80
F. Wilt	4.80	9.50	78.60	3.5	81.54
Healthy	2.2	0.7	0.5	77.70	95.81
	85.33	86.31	93.57	91.41	89.10

Figure 9. Confusion matrices of Cotton dataset

DarkNet-19

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	65.40	19.50	8.40	70.10
Bean Rust	13.00	59.30	8.80	73.12
Healthy	7.60	8.20	67.80	81.10
	76.05	68.16	79.76	74.61

GoogleNet

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	52.70	22.00	8.40	63.42
Bean Rust	19.60	52.30	13.70	61.10
Healthy	13.70	12.70	62.90	70.44
	61.28	60.11	74.00	65.08

Inception-v3

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	53.40	15.60	7.30	69.99
Bean Rust	20.10	61.40	15.00	63.63
Healthy	12.50	10.00	62.70	73.59
	62.09	70.57	73.76	68.80

ResNet-18

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	64.90	20.90	9.10	68.39
Bean Rust	14.40	56.00	12.80	67.31
Healthy	6.70	10.10	63.10	78.97
	75.47	64.37	74.24	71.32

ShuffleNet

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	49.90	18.50	8.30	65.06
Bean Rust	25.80	59.70	22.80	55.12
Healthy	10.30	8.80	53.90	73.84
	58.02	68.62	63.41	63.37

Pretrained DarkNet-19

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	82.50	5.90	1.70	91.56
Bean Rust	3.00	79.50	2.20	93.86
Healthy	0.50	1.60	81.10	97.48
	95.93	91.38	95.41	94.22

Pretrained GoogleNet

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	79.80	8.40	0.90	89.56
Bean Rust	6.20	76.10	4.30	87.88
Healthy	0.00	2.50	79.80	96.96
	92.79	87.47	93.88	91.36

Pretrained Inception-v3

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	77.00	3.20	0.10	95.89
Bean Rust	7.20	80.90	1.90	89.89
Healthy	1.80	2.90	83.00	94.64
	89.53	92.99	97.65	93.37

Pretrained ResNet-18

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	80.20	2.80	0.70	95.82
Bean Rust	5.60	83.20	3.40	90.24
Healthy	0.20	1.00	80.90	98.54
	93.26	95.63	95.18	94.69

Pretrained ShuffleNet

	A. L. Spot	Bean Rust	Healthy	
A. L. Spot	78.10	5.50	0.70	92.65
Bean Rust	6.90	78.50	4.00	87.81
Healthy	1.00	3.00	80.30	95.26
	90.81	90.23	94.47	91.82

Figure 10. Confusion matrices of Bean dataset

DarkNet-19

	Cor.	Healthy	Pest.	Siga.	
Cor.	31.40	0.10	1.00	0.7	94.58
Healthy	0.00	24.80	9.50	4.4	64.08
Pest.	0.50	1.00	20.00	1	88.89
Siga.	0.1	0.1	4.5	88.90	94.98
	98.13	95.38	57.14	93.58	87.82

GoogleNet

	Cor.	Healthy	Pest.	Siga.	
Cor.	24.60	7.00	13.40	7.5	46.86
Healthy	3.40	12.30	5.00	0	59.42
Pest.	2.70	6.10	9.20	0.3	50.27
Siga.	1.3	0.6	7.4	87.20	90.36
	76.88	47.31	26.29	91.79	70.90

Inception-v3

	Cor.	Healthy	Pest.	Siga.	
Cor.	27.80	0.40	0.30	2.5	89.68
Healthy	0.20	20.70	11.10	2.5	60.00
Pest.	1.60	4.10	15.50	0.5	71.43
Siga.	2.4	0.8	8.1	89.50	88.79
	86.88	79.62	44.29	94.21	81.65

ResNet-18

	Cor.	Healthy	Pest.	Siga.	
Cor.	30.70	0.20	1.10	2.2	89.77
Healthy	0.00	21.70	10.30	0	67.81
Pest.	0.20	3.80	15.90	0.1	79.50
Siga.	1.1	0.3	7.7	92.70	91.06
	95.94	83.46	45.43	97.58	85.64

ShuffleNet

	Cor.	Healthy	Pest.	Siga.	
Cor.	21.90	0.70	1.60	2.6	81.72
Healthy	0.00	20.30	13.80	3	54.72
Pest.	1.20	3.90	11.50	0.3	68.05
Siga.	8.9	1.1	8.1	89.10	83.12
	68.44	78.08	32.86	93.79	75.96

Pretrained DarkNet-19

	Cor.	Healthy	Pest.	Siga.	
Cor.	31.50	0.10	0.80	1.3	93.47
Healthy	0.30	25.10	3.30	4.8	74.93
Pest.	0.00	0.80	26.20	3.8	85.06
Siga.	0.2	0	4.7	85.10	94.56
	98.44	96.54	74.86	89.58	89.31

Pretrained GoogleNet

	Cor.	Healthy	Pest.	Siga.	
Cor.	32.00	0.00	0.80	7.4	79.60
Healthy	0.00	26.00	0.50	0.4	96.65
Pest.	0.00	0.00	31.90	1.7	94.94
Siga.	0	0	1.8	85.50	97.94
	100.00	100.00	91.14	90.00	93.30

Pretrained Inception-v3

	Cor.	Healthy	Pest.	Siga.	
Cor.	31.80	0.00	0.60	2.6	90.86
Healthy	0.10	26.00	1.10	0.5	93.86
Pest.	0.00	0.00	29.90	7.8	79.31
Siga.	0.1	0	3.4	84.10	96.00
	99.38	100.00	85.43	88.53	91.38

Pretrained ResNet-18

	Cor.	Healthy	Pest.	Siga.	
Cor.	32.00	0.00	0.80	1.9	92.22
Healthy	0.00	26.00	1.80	0.5	91.87
Pest.	0.00	0.00	28.90	9.2	75.85
Siga.	0	0	3.5	83.40	95.97
	100.00	100.00	82.57	87.79	90.59

Pretrained ShuffleNet

	Cor.	Healthy	Pest.	Siga.	
Cor.	31.90	0.00	2.70	2.1	86.92
Healthy	0.00	26.00	2.40	1.5	86.96
Pest.	0.10	0.00	25.60	7.1	78.05
Siga.	0	0	4.3	84.30	95.15
	99.69	100.00	73.14	88.74	89.26

Figure 11. Confusion matrices of Banana dataset

The Receiver-Operating Characteristic (ROC) curves obtained from potato, cotton, bean, and banana datasets are shown in Figures 12, 13, 14, and 15, respectively. ROC curves show the relationship between the false-positive rate (FPR) and the true-positive rate (TPR). It is seen that pretrained CNN models have higher true-positive rates in all datasets.
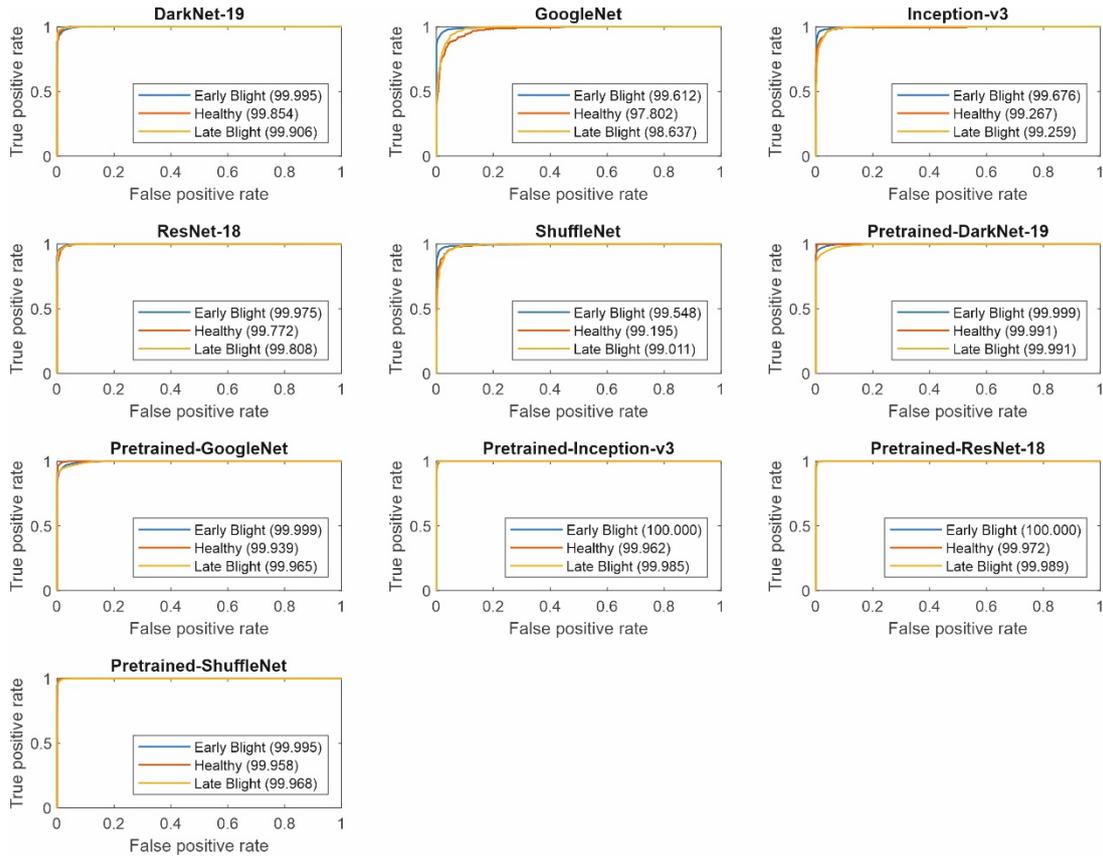
Figure 12. ROC Curves for Potato dataset
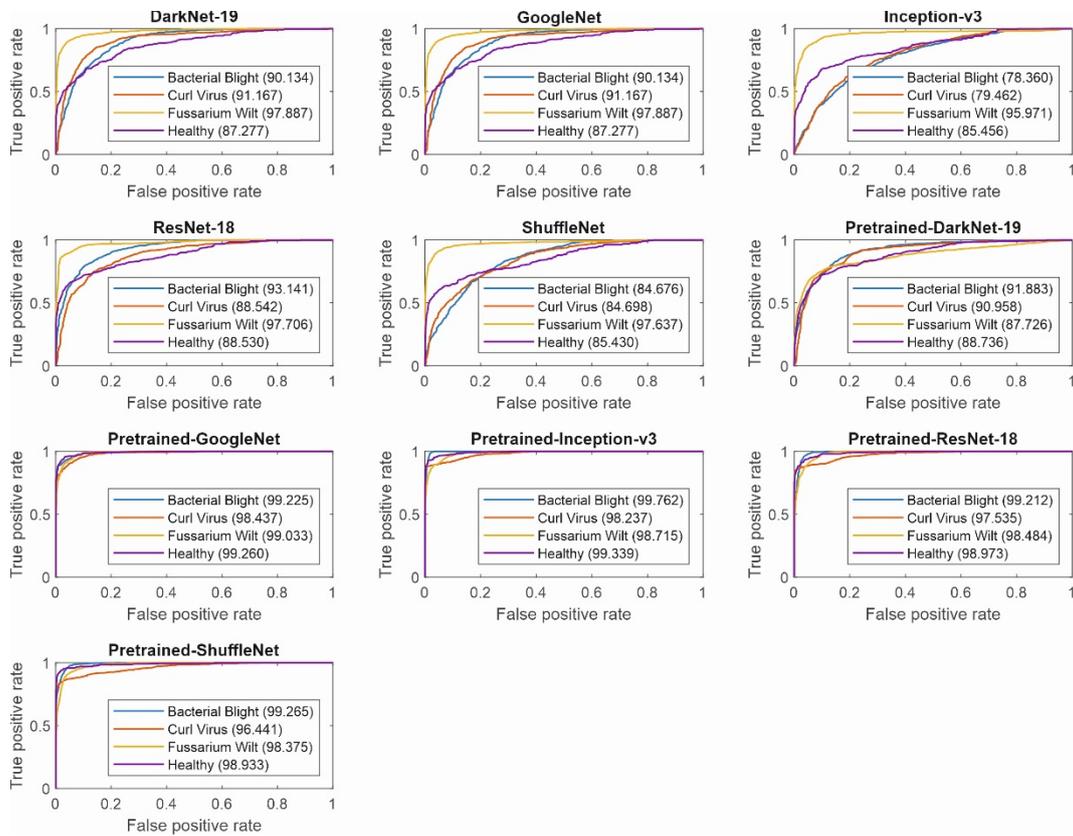
Figure 13. ROC Curves for Cotton dataset
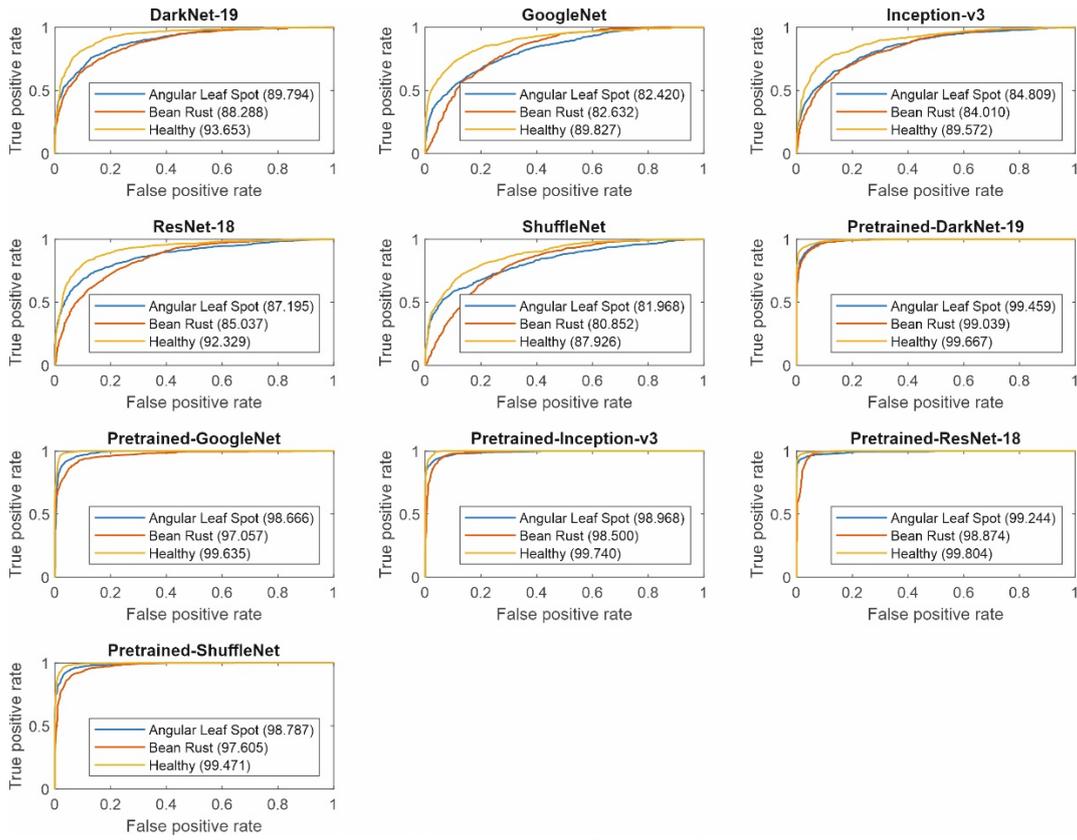
Figure 14. ROC Curves for Bean dataset
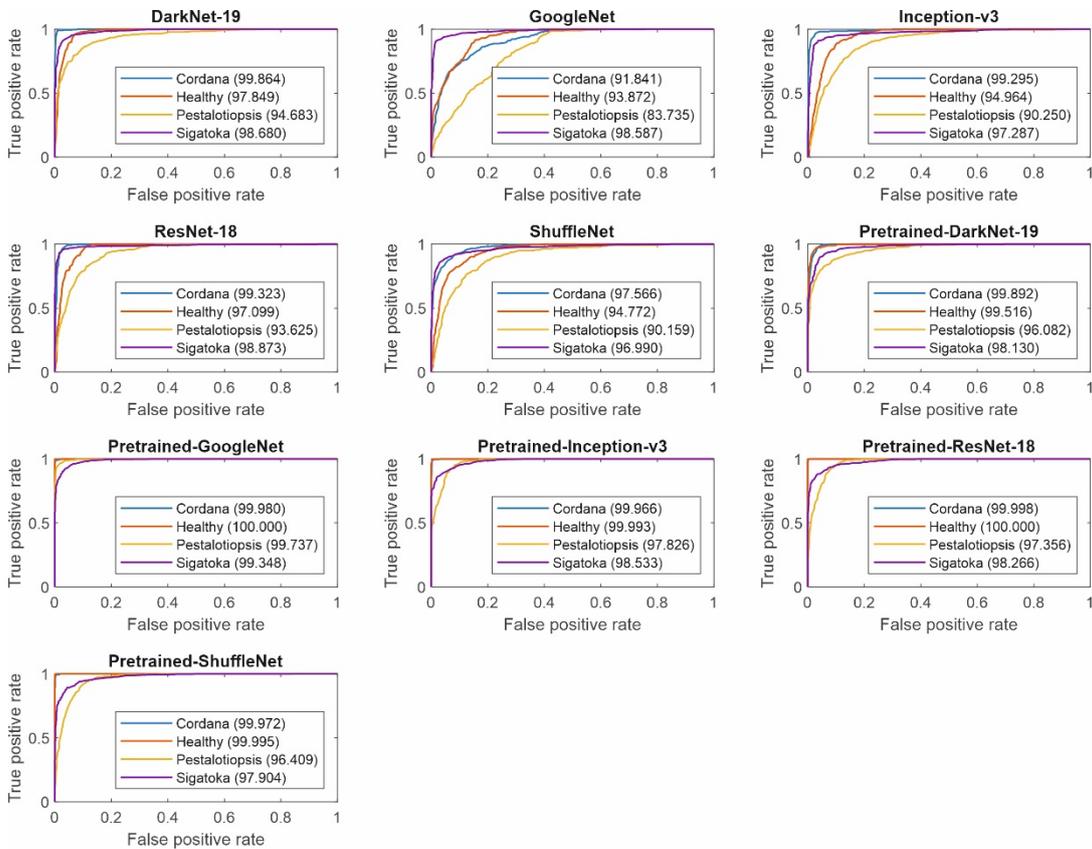
Figure 15. ROC Curves for Banana dataset

4. CONCLUSIONS

Plant diseases lead to decreases in crop productivity, economic losses, and an inability to meet the increasing demand. However, early detection of plant diseases plays an important role in preventing these losses. Methods based on image processing technology play an important role in the detection of plant diseases. In this study, DarkNet-19, GoogleNet, Inception-v3, Resnet-18 and, ShuffleNet models, which are popular CNN models, have been used for automatic classification of diseases from leaf images of potato, banana, cotton, and bean. In addition, pretrained versions of these models, the weights of which were previously trained in different large databases with the "Transfer Learning" method, have also been used in the study. When the experiment results were examined in general, it was seen that the Pretrained-DarkNet-19 and Pretrained-GoogleNet models had the highest performance. In addition, the experiment results revealed that the success rates of standard CNN models increase by 7% to 25% with the transfer learning method, even at very low numbers such as 5 epochs. In future studies, features can be extracted from plant disease images with CNN and various classification algorithms can be used in the classification phase.

REFERENCES

[1] Türkoğlu, M., Hanbay, K., Sivrikaya, I. S., and Hanbay, D. (2021). Derin Evrişimsel Sinir Ağı Kullanılarak Kayısı Hastalıklarının Sınıflandırılması, *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 9(1): 334-345.

[2] Aslan, M. (2021). Derin Öğrenme ile Bitki Hastalıklarının Tespiti, *Avrupa Bilim ve Teknoloji Dergisi*, 23: 540-546.

[3] Chen, J., Zhang, D., Zeb, A., and Nanehkaran, Y. A. (2021). Identification of rice plant diseases using lightweight attention network, *Expert Systems with Applications*, 169: 114514.

[4] Shrivastava, V. K., Pradhan, M. K., Minz, S., and Thakur, M. P. (2019). Rice plant disease classification using transfer learning of deep convolution neural network, *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 3(6): 631–635.

[5] Abbas, A., Jain, S., Gour, M., and Vankudothu, S. (2021). Tomato plant disease detection using transfer learning with C-GAN synthetic images, *Computers and Electronics in Agriculture*, 187: 106279.

[6] Sert, E. (2021). A deep learning based approach for the detection of diseases in pepper and potato leaves, *Anadolu Tarım Bilimleri Dergisi*, 36(2): 167-178.

[7] Khan, M. A., Akram, T., Sharif, M., Awais, M., Javed, K., Ali, H., and Saba, T. (2018). CCDF: Automatic system for segmentation and recognition of fruit crops diseases based on correlation coefficient and deep CNN features, *Computers and electronics in agriculture*, 155: 220-236.

[8] Aksoy, B., Halis, H. D., and Salman, O. K. M. (2020). Elma Bitkisindeki Hastalıkların Yapay Zekâ Yöntemleri ile Tespiti ve Yapay Zekâ Yöntemlerinin Performanslarının Karşılaştırılması, *International Journal of Engineering and Innovative Research*, 2(3): 194-210.

[9] Hassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z., and Jasińska, E. (2021). Identification of Plant-Leaf Diseases Using CNN and Transfer-Learning Approach, *Electronics*, 10(12): 1388.

[10] Arivazhagan, S., and Ligi, S. V. (2018). Mango leaf diseases identification using convolutional neural network, *International Journal of Pure and Applied Mathematics*, 120(6): 11067-11079.

[11] Priyadharshini, R. A., Arivazhagan, S., Arun, M., and Mirnalini, A. (2019). Maize leaf disease classification using deep convolutional neural networks, *Neural Computing and Applications*, 31(12): 8887-8895.

[12] Dinata, M. I., Nugroho, S. M. S., and Rachmadi, R. F. (2021). Classification of Strawberry Plant Diseases with Leaf Image Using CNN, *International Conference on Artificial Intelligence and Computer Science Technology (ICAICST) IEEE*, Yogyakarta, Indonesia, 68-72.

[13] Qin, Z., Yu, F., Liu, C., and Chen, X. (2018). How convolutional neural network see the world-A survey of convolutional neural network visualization methods. *arXiv preprint* arXiv:1804.11191.

[14] Redmon, J., and Farhadi, A. (2017). YOLO9000: better, faster, stronger. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, HI, USA, 7263-7271.

[15] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D. and Rabinovich, A. (2015). Going deeper with convolutions, *In Proceedings of the IEEE conference on computer vision and pattern recognition*, Boston, USA, 1-9.

[16] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision, *In Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, USA, 2818-2826.

[17] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition, *In Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, USA, 770-778.

[18] Zhang, X., Zhou, X., Lin, M., and Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices, *In Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City, USA, 6848-6856.

[19] Fırıldak, K., and Talu, M. F. (2019). Evrişimsel Sinir Ağlarında Kullanılan Transfer Öğrenme Yaklaşımlarının İncelenmesi, *Computer Science*, 4(2): 88-95.

[20] Nath, S. (2021). Potato Leaf Disease Detection. Kaggle Data, version 1. Retrieved September 20, 2021 from https://www.kaggle.com/sayannath235/potato-leaf-disease-detection/metadata.

[21] Karim, S. (2021). Cotton Leaf Disease Dataset. Kaggle Data, version 1. Retrieved September 20, 2021 from https://www.kaggle.com/seroshkarim/cotton-leaf-disease-dataset/metadata.

[22] Rastogi, P. (2021). Bean leaf dataset. Kaggle Data, version 1. Retrieved September 20, 2021 from https://www.kaggle.com/prakharrastogi534/bean-leaf-dataset/metadata.

[23] Mahmud, K. A. (2021). Banana Leaf Dataset. Kaggle Data, version 2. Retrieved September 20, 2021 from https://www.kaggle.com/kaiesalmahmud/banana-leaf-dataset.